Af
IFW

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re application of:

MICHAEL S. MILILLO ET AL.

Group Art Unit: 2185

Examiner: John M. Ross

Serial No.: 10/023,321

Filed: December 18, 2001

For: CACHE STORAGE SYSTEM AND METHOD

Attorney Docket No.: 2001-053-ICE

APPEAL BRIEF UNDER 37 C.F.R. § 41.37

Mail Stop Appeal Brief - Patents
Commissioner for Patents
U.S. Patent & Trademark Office
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

This is an Appeal Brief from the final rejection of claims 1, 4-6 and 9-22 of the Office Action mailed on May 11, 2005, for the above-identified patent application.

I. REAL PARTY IN INTEREST

The real party in interest is Storage Technology Corporation ("Assignee"), a corporation organized and existing under the laws of the state of Delaware, and having a place of business at One StorageTek Drive, MS-4309, Louisville, Colorado 80028-4309, as set forth in the assignment recorded in the U.S. Patent and Trademark Office on December 18, 2001 at Reel 012396/Frame 0110.

CERTIFICATE OF MAILING UNDER 37 C.F.R. § 1.8

I hereby certify that this paper, including all enclosures referred to herein, is being deposited with the United States Postal Service as first-class mail, postage pre-paid, in an envelope addressed to: Mail Stop Appeal Brief - Patents, Commissioner for Patents, U.S. Patent & Trademark Office, P.O. Box 1450, Alexandria, VA 22313-1450 on:

November 23, 2005	Timothy J. Marsh	Timothy J. Marsh
Date of Deposit	Name of Person Signing	Signature

II. RELATED APPEALS AND INTERFERENCES

There are no appeals or interferences known to the Appellant, the Appellant's legal representative, or the Assignee which will directly affect or be directly affected by or have a bearing on the Board's decision in the pending appeal.

III. STATUS OF CLAIMS

Claims 1, 4-6 and 9-22 are pending in this application. Claims 1, 4-6 and 9-22 have been rejected and are the subject of this appeal.

IV. STATUS OF AMENDMENTS

An amendment after final rejection was filed on September 9, 2005, and has been accepted for entry.

V. SUMMARY OF CLAIMED SUBJECT MATTER

The invention relates to a cache storage system for use in a data storage system having a plurality of virtual addresses. Each virtual address has a data object associated therewith. The cache storage system comprises a plurality of storage devices (12), each data object being stored at a storage device location (24). Each storage device location (24) has a unique identifier. The cache storage system further comprises a cache (32) for storing a data object associated with at least one virtual address. For a first virtual address, the first virtual address data object is staged into the cache (32). For a second virtual address, a pointer (36ii) is generated for use in pointing to the first virtual address data object staged in the cache (32) when the storage device location identifier of the second virtual address data object matches the storage device location identifier of the first virtual address data object. The cache (32) comprises a location identifier table (42) for storing at least one storage device location identifier, and a virtual address table (44) for storing a plurality of virtual addresses. (See, for example, FIG. 1, 3, 6 and Specification at page 10 line 1 through page 12 line 5).

Another expression of the invention relates to a cache storage method for use in a data storage system comprising a plurality of storage devices (12) and having a plurality of virtual addresses, each virtual address having a data object associated therewith. The method comprises storing each data object at a storage device location (24), each storage device location (24) having a unique identifier. The method further comprises storing in a cache (32) a data object associated with at least one virtual address wherein, for a first virtual address, the first virtual address data object is staged into the cache (32) and, for a second virtual address, a pointer (36ii) is generated for use in pointing to the first virtual address data object staged in the cache (32) when the storage device location identifier of the second virtual address data object matches the storage device location identifier of the first virtual address data object. The cache (32) comprises a location identifier table (42) for storing at least one storage device location identifier, and a virtual address table (44) for storing a plurality of virtual addresses. (See, for example, FIG. 1, 3, 6 and Specification at page 14 line 30 through page 15 line 23).

Yet another expression of the invention relates to a cache storage system for use in a data storage system. The data storage system comprises a plurality of storage devices (12) and has a plurality of virtual addresses. Each virtual address is associated with a data object. Each data object is stored at a storage device location (24). Each storage device location (24) has a unique identifier. The cache storage system comprises a cache (32) for storing a data object associated with at least one virtual address, a virtual address table (44) for storing a plurality of virtual addresses, and a location identifier table (42) for storing at least one storage device location identifier. For a first virtual address, the first virtual address data object is staged into the cache (32), the location identifier for the first virtual address data object is stored in the location identifier table (42), and the first virtual address is stored in the virtual address table (44) and linked to the location identifier for the first virtual address data object stored in the location identifier table (42). For a second virtual address, a pointer (36ii) is generated for use in pointing to the first virtual address data object staged in the cache (32)

when the location identifier of the second virtual address data object matches the location identifier stored in the location identifier table (42) of the first virtual address data object, and the second virtual address is stored in the virtual address table (44) and linked to the first virtual address. (See, for example, FIG. 1, 3, 7 and Specification at page 10 line 1 through page 12 line 5).

Still yet another expression of the invention relates to a cache storage method for use in a data storage system. The data storage system comprising a plurality of storage devices (12) and has a plurality of virtual addresses. Each virtual address is associated with a data object. Each data object is stored at a storage device location (24). Each storage device location (24) has a unique identifier. The method comprises storing in a cache (32) a data object associated with at least one virtual address, storing in a virtual address table (44) a plurality of virtual addresses, and storing in a location identifier table (42) at least one storage device location identifier. For a first virtual address, the first virtual address data object is staged into the cache (32), the location identifier for the first virtual address data object is stored in the location identifier table (42), and the first virtual address is stored in the virtual address table (44) and linked to the location identifier for the first virtual address data object stored in the location identifier table (42). For a second virtual address, a pointer is generated (36ii) for use in pointing to the first virtual address data object staged in the cache (32) when the location identifier of the second virtual address data object matches the location identifier stored in the location identifier table (42) of the first virtual address data object, and the second virtual address is stored in the virtual address table (44) and linked to the first virtual address. (See, for example, FIG. 1, 3, 7 and Specification at page 15 line 24 through page 16 line 24).

VI. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

Claims 1, 4-6 and 9-22 stand rejected under 35 U.S.C. § 102(b) as being anticipated by U.S. Patent No. 5,379,391 to Belsan et al.

VII. ARGUMENT

A. Claims 1, 4-6 and 9-22 Are Patentable Under 35 U.S.C. § 102(b) Over U.S. Patent No. 5,379,391 to Belsan et al.

The Examiner has rejected claims 1, 4-6 and 9-22 under 35 U.S.C. § 102(b) as being anticipated by commonly owned U.S. Patent No. 5,379,391 to Belsan et al. (hereinafter "Belsan"). The Examiner has failed to establish, however, that Belsan discloses all of the limitations of the Applicant's claimed invention. As noted in § 2131 of the *Manual of Patent Examining Procedure*, "a claim is anticipated only if each and every element as set forth in the claim is found, either expressly or inherently described, in a single prior art reference." *Verdegaal Bros. v. Union Oil Co. of California*, 814 F.2d 628, 631 2 USPQ2d 1051, 1053 (Fed. Cir. 1987). Similarly, "the identical invention must be shown in as complete detail as is contained in the ... claim." *Richardson v. Suzuki Motor Co.*, 868 F.2d 1226, 1236 , 9 USPQ2d 1913, 1920 (Fed. Cir. 1989).

Independent claims 1 and 6 require that for a second virtual address, a pointer is generated for use in pointing to the first virtual address data object staged in the cache when the storage device location identifier of the second virtual address data object matches the storage device location identifier of the first virtual address data object. Similarly, independent claims 11 and 17 require that for a second virtual address, a pointer is generated for use in pointing to the first virtual address data object staged in the cache when the location identifier of the second virtual address data object matches the location identifier stored in the location identifier table of the first virtual address data object.

In contrast to the present invention, Belsan discloses a move/copy operation that instantaneously relocates or creates a second instance of a selected data file by merely generating a new set of pointers to reference the same physical memory location as the original

set of reference pointers in the virtual track directory. Belsan further discloses that the apparatus instantaneously moves the original data file without the time penalty of having to down-load the data file to the cache memory and write the data file to a new physical memory location. (See Belsan, col. 8 lines 31-43, Abstract). Accordingly, Belsan simply does not disclose a pointer generated for use in pointing to the first virtual address data object staged in the cache when the storage device location identifier of the second virtual address data object matches the storage device location identifier of the first virtual address data object. Because Belsan does not disclose all of the limitations of the Applicant's claimed invention, the rejection should be reversed.

The Examiner appears to rely on Belsan's disclosure of cache consistency protection to teach the missing limitations (See Advisory Action mailed October 5, 2005, reference nos. 4 and 5, Office Action mailed May 11, 2005 reference no. 10 lines 10-14). However, Applicant respectfully contends that the Examiner has misapplied the teachings of Belsan. Cache consistency protection, as disclosed by Belsan, is directed to a scenario wherein a host processor modifies a data record that has been staged into cache memory from a disk drive subset. The modification of the record in cache memory renders the modified record inconsistent with the original data record on the disk drive subset. To remedy the potential data inconsistency, Belsan teaches that all of the virtual addresses stored in the mapping tables and associated with the original data record are made available when the original data record is loaded into cache memory. These virtual addresses are, for example, all loaded into the hash table of cache memory. If an associated host processor attempts to update that data record at one of these virtual addresses, the virtual memory system copies the data record to a new cache memory location so that a unique instance of the data record is available for update while the remaining virtual addresses still point to the original data record in cache memory. The virtual address used by the host processor to address the now modified data record is expunged from the copy table and assigned to the new cache memory location. (See Belsan, col. 3 lines 10-31, col. 13 line 41 through col. 14 line 7).

Accordingly, Belsan discloses virtual addresses associated with an original data record that are loaded into a hash table when the original data record is loaded into cache memory. After the virtual addresses are loaded into the hash table, Belsan simply teaches the re-assignment of a virtual address associated with a modification to the data record. Accordingly, the operation of cache consistency protection may result in the removal (i.e., expungement) of a virtual address from the copy table but does not disclose or otherwise provide a pointer generated for use in pointing to a first virtual address data object staged in the cache, as presently claimed. As such, Belsan does not disclose all of the limitations of the Applicant's claimed invention and the rejection should be reversed.

Furthermore, the data read and data write operations disclosed by Belsan are inconsistent with the generation of a pointer pointing to a first virtual address data object staged in the cache when the storage device location identifier of a second virtual address data object matches the storage device location identifier of the first virtual address data object.

In particular, Belsan discloses a data read operation wherein the control unit branches to a cache directory search subroutine to determine whether the virtual track to be read is already staged in the cache memory. The cache directory search subroutine includes the step of scanning the hash table directory of the cache memory to determine whether the requested virtual track is located in the cache memory. If it is, the data from the staged virtual track instance is transferred. If not, the control unit looks up the address of the virtual track in the virtual to logical map table. The logical map location is used to map the logical device to one or more physical devices. The control unit then schedules one or more physical read operations to retrieve the virtual track instance from the identified physical devices. (See Belsan, col. 11 lines 1-42 and FIG. 6).

Similarly, Belsan discloses a data write operation wherein the control unit branches to the cache directory search subroutine to assure that the virtual track into which the

data is to be transferred is located in the cache memory. Since all of the data updating is performed in the cache memory, the virtual track in which the data is to be written must be transferred to the cache memory if it is not already resident in the cache memory. If the virtual track is not already in the cache memory, the transfer of the virtual track instance to the cache memory is performed identically to the data read operation. That is, the control unit looks up the address of the virtual track in the virtual to logical map table. The logical map location is used to map the logical device to one or more physical devices. The control unit then schedules one or more physical read operations to retrieve the virtual track instance from the identified physical devices.(See Belsan, col. 12 lines 42-55 and FIG. 7).

As such, the read and write operations of Belsan reference the logical and physical addresses of the virtual track only as a precursor to retrieving the virtual track from an appropriate physical device. It is clear, therefore, that Belsan does not disclose the use of the logical or physical addresses to identify when the storage device location identifier of a second virtual address data object matches the storage device location identifier of a first virtual address data object staged in cache in order to generate a pointer for use in pointing to the first virtual address data object.

For the forgoing reasons, Belsan does not disclose all of the limitations of the Applicant's claimed invention and the rejection of independent claims 1, 6, 11 and 17 should be reversed. Claims 4-5, 9-10, 12-16 and 18-22 depend from independent claim 1, 6, 11 and 17, respectively. Therefore, Applicant respectfully requests that the rejection of these claims be reversed for the reasons the rejection of the independent claims should be reversed.

CONCLUSION

For the reasons set forth above, the Examiner has failed to establish a *prima facie* case for the rejection of claims 1, 4-6 and 9-22. Therefore, the final rejection of these claims should be reversed.

Please charge the fee of $500.00 as applicable under the provisions of 37 C.F.R. § 41.20(b)(2) and any additional fees or credit any overpayment in connection with this filing to Deposit Account No. 19-4545.

Respectfully submitted,

MICHAEL S. MILILLO ET AL.

By: Timothy J. Marsh

Timothy J. Marsh
Registration No. 56,684
Attorney/Agent for Applicant

Date: November 23, 2005

BROOKS KUSHMAN P.C.
1000 Town Center, 22nd Floor
Southfield, MI 48075-1238
Phone: 248-358-4400
Fax: 248-358-3351

Enclosure - Appendices

VIII. CLAIMS APPENDIX

1. A cache storage system for use in a data storage system having a plurality of virtual addresses, each virtual address having a data object associated therewith, the cache storage system comprising:

a plurality of storage devices, each data object being stored at a storage device location, each storage device location having a unique identifier; and

a cache for storing a data object associated with at least one virtual address wherein, for a first virtual address, the first virtual address data object is staged into the cache and, for a second virtual address, a pointer is generated for use in pointing to the first virtual address data object staged in the cache when the storage device location identifier of the second virtual address data object matches the storage device location identifier of the first virtual address data object, and wherein the cache comprises a location identifier table for storing at least one storage device location identifier, and a virtual address table for storing a plurality of virtual addresses.

4. The system of claim 1 wherein the data storage system comprises a disk subsystem, the plurality of storage devices comprise a plurality of disk storage devices, each virtual address comprises a virtual track address, and each storage device location identifier comprises a track number.

5. The system of claim 1 wherein the pointer comprises an entry in a cache directory, the cache directory entry comprising a location in the cache of a segment storing data associated with a data object shared by the first and second virtual addresses.

6. A cache storage method for use in a data storage system comprising a plurality of storage devices and having a plurality of virtual addresses, each virtual address having a data object associated therewith, the method comprising:

storing each data object at a storage device location, each storage device location having a unique identifier; and

storing in a cache a data object associated with at least one virtual address wherein, for a first virtual address, the first virtual address data object is staged into the cache and, for a second virtual address, a pointer is generated for use in pointing to the first virtual address data object staged in the cache when the storage device location identifier of the second virtual address data object matches the storage device location identifier of the first virtual address data object, and wherein the cache comprises a location identifier table for storing at least one storage device location identifier, and a virtual address table for storing a plurality of virtual addresses.

9. The method of claim 6 wherein the data storage system comprises a disk subsystem, the plurality of storage devices comprise a plurality of disk storage devices, each virtual address comprises a virtual track address, and each storage device location identifier comprises a track number.

10. The method of claim 6 wherein the pointer comprises an entry in a cache directory, the cache directory entry comprising a location in the cache of a segment storing data associated with a data object shared by the first and second virtual addresses.

11. A cache storage system for use in a data storage system, the data storage system comprising a plurality of storage devices and having a plurality of virtual addresses, each virtual address associated with a data object, each data object stored at a storage device location, each storage device location having a unique identifier, the cache storage system comprising:

a cache for storing a data object associated with at least one virtual address;

a virtual address table for storing a plurality of virtual addresses; and

a location identifier table for storing at least one storage device location identifier;

wherein, for a first virtual address, the first virtual address data object is staged into the cache, the location identifier for the first virtual address data object is stored in the location identifier table, and the first virtual address is stored in the virtual address table and linked to the location identifier for the first virtual address data object stored in the location identifier table, and wherein, for a second virtual address, a pointer is generated for use in pointing to the first virtual address data object staged in the cache when the location identifier of the second virtual address data object matches the location identifier stored in the location identifier table of the first virtual address data object, and the second virtual address is stored in the virtual address table and linked to the first virtual address.

12. The system of claim 11 wherein the virtual address table is stored in the cache.

13. The system of claim 11 wherein the location identifier table is stored in the cache.

14. The system of claim 11 wherein the virtual address and location identifier tables are stored in the cache.

15. The system of claim 11 wherein the data storage system comprises a disk subsystem, the plurality of storage devices comprises a plurality of disk storage devices, each virtual address comprises a virtual track address, each storage device location identifier comprises a track number, the virtual address table comprises a virtual track number table, and the location identifier table comprises a track number table.

16. The system of claim 11 wherein the pointer comprises an entry in a cache directory, the cache directory entry comprising a location in the cache of a segment storing data associated with a data object shared by the first and second virtual addresses.

17. A cache storage method for use in a data storage system, the data storage system comprising a plurality of storage devices and having a plurality of virtual addresses, each virtual address associated with a data object, each data object stored at a storage device location, each storage device location having a unique identifier, the method comprising:

storing in a cache a data object associated with at least one virtual address;

storing in a virtual address table a plurality of virtual addresses; and

storing in a location identifier table at least one storage device location identifier;

wherein, for a first virtual address, the first virtual address data object is staged into the cache, the location identifier for the first virtual address data object is stored in the location identifier table, and the first virtual address is stored in the virtual address table and linked to the location identifier for the first virtual address data object stored in the location identifier table, and wherein, for a second virtual address, a pointer is generated for use in pointing to the first virtual address data object staged in the cache when the location identifier of the second virtual address data object matches the location identifier stored in the location identifier table of the first virtual address data object, and the second virtual address is stored in the virtual address table and linked to the first virtual address.

18. The method of claim 17 wherein the virtual address table is stored in the cache.

19. The method of claim 17 wherein the location identifier table is stored in the cache.

20. The method of claim 17 wherein the virtual address and location identifier tables are stored in the cache.

21. The method of claim 17 wherein the data storage system comprises a disk subsystem, the plurality of storage devices comprise a plurality of disk storage devices, each virtual address comprises a virtual track address, each storage device location identifier comprises a track number, the virtual address table comprises a virtual track number table, and the location identifier table comprises a track number table.

22. The method of claim 17 wherein the pointer comprises an entry in a cache directory, the cache directory entry comprising a location in the cache of a segment storing data associated with a data object shared by the first and second virtual addresses.

IX. EVIDENCE APPENDIX

None

X. RELATED PROCEEDINGS APPENDIX

None